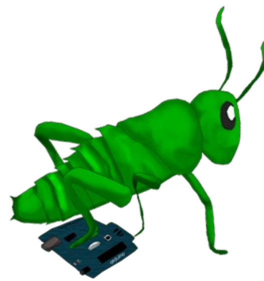
Proyecto Cañafote: Redes de sensores
Manual de instalación



Version 0.1 19/04/2010

Álvaro Neira Ayuso <alvaroneay@gmail.com>

Paso 0 : Dependencias que tiene el software

En este apartado voy a explicaros que es lo que necesitáis para que podáis echar ha andar este programa. Los programas que debemos instalar antes de que empecemos a montar todo el proyecto, son:

1-. El módulo de Python, ya que todos los programas que son utilizados en el ordenador (Programa de escucha, ParserMedia, etc) están programados en Python y no podrán funcionar si no tenemos el compilador de este lenguaje la verdad que poco vamos ha hacer. Podéis descargarlo desde la su web, http://www.python.org/download/ podéis buscar el programa para vuestro sistema operativo (si usáis linux y no lo tenéis instalado el paquete está en el repositorios).

2-. Es necesario el uso de la librería de Mathplotlib para generar las gráficas con los datos que envían los arduinos al servidor. Esta librería la podemos obtener desde su web http://sourceforge.net/projects/matplotlib/files/matplotlib/matplotlib-0.99.1/ y os lo encontraréis para cualquier sistema operativo (desde linux igual que python, lo podéis encontrar ya que esta en los repositorios).

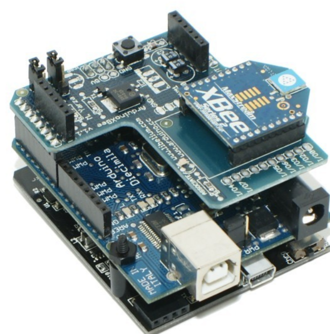
3-. Este es nuestro ultimo programa que necesitamos instalar. El Mathplotlib tiene dependencia de un programa para que funcione y es el Numpy, también podéis obtenerlo desde la web que os adjunto http://new.scipy.org/download.html viene también para todas las plataformas.

Ya después de instalar estos tres programas, ya podemos ponernos manos a la obra para echar ha andar el programa del proyecto Cañafote.

Paso 1: Montaje Físico de los arduinos con los sensores

Lo primero que debemos hacer es ensamblar los respectivos módulos, la verdad que no esté paso no me voy a parar mucho tiempo ya que es un poco lógico, el módulo de ethernet para ensamblarlo tiene que encajar todo los pines con todo los zócalos del arduino (o sea sé, solo hay una forma para colocarlo) y para el módulo Xbee solo puedes encajarlo de una manera si queremos que todos los pines estén conectados con el arduino.

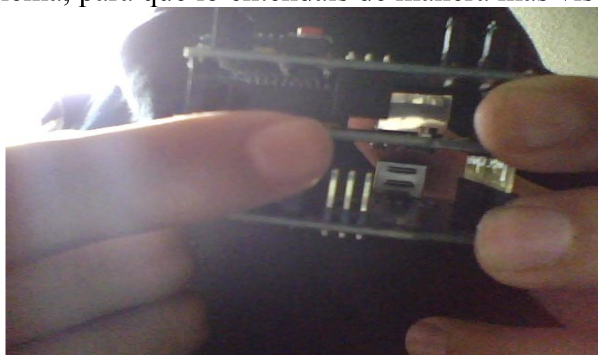




El módulo Ethernet está forma   Así queda el módulo Ethernet

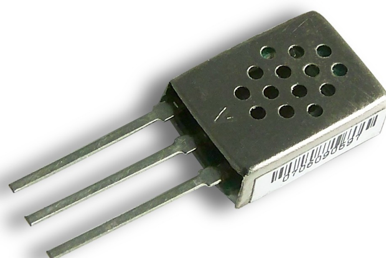
El único problema que nos encontramos es cuando tenemos que colocar el módulo Xbee con el Ethernet para hacer el receptor Xbee, ya que en el programa tenemos que enganchar el arduino al módulo Ethernet y después al módulo Xbee, ya que el Xbee se nos queda sin alimentación,

os adjunto la foto del problema, para que lo entendáis de manera más visual.



Los pines de abajo de mi dedo, deberían estar conectados al zócalo de arriba, yo lo que he hecho es existen unos cables que vienen de dos en dos, siendo unos pequeños zócalos para conectar los pines de abajo y pines para conectarlos arriba, ya tenemos solucionado nuestro problema.
Ya montado los módulos vamos ha pasar al montaje de los sensores, no tiene complicación pero si es un poco lioso, así que voy a ir paso a paso. El primer sensor que vamos a montar es el sensor de humedad.

El sensor de humedad es Sensor de Humedad 808H5V5 comprado en www.libelium.com/tienda/catalog y es igual a esté que aparece en la siguiente foto.

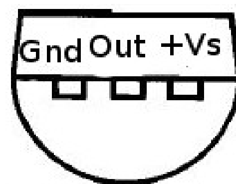


Donde la patilla de la izquierda de la V es la patilla de GND o tierra, la central (la que tiene la V) es

la que nos da la señal de salida que es la que va a leer el arduino como una señal Analógica (da un rango de 0 a 1024) y la patilla de la derecha es la de alimentación donde tiene que entrar 5 voltios para que el sensor comience a funcionar. Solo falta que hagáis las conexiones siguiente la explicación de cada patilla, conectad la patilla izquierda del todo a tierra, la centrar al zócalo de señales analógicas número 4, ya que en el programa del proyecto tiene programado que el zócalo número 4 de las señales analógicas sea leído para obtener la humedad.
Ya con esto tenemos montado el sensor de humedad, ahora nos falta montar el sensor de temperatura, el nombre del sensor de temperatura es el LM35 también comprado desde la página web de libelium que os he dado antes, os adjunto una foto.



Os encontraréis un montón de datasheet, incluso el del fabricante, han tenido un despiste ya que está mal organizado y si lo sigues solamente puede pasar que te cargues el sensor y además te quemes los dedos como me pasó a mí... Os adjunto un datasheet que me he hecho yo, para que sigáis mis este esquema que os doy.



Ya tenemos este esquema, tenéis que hacer las mismas conexiones como con el sensor de humedad pero hay una diferencia y es que este sensor su salida para el arduino y es demasiado grande y cuando hace la lectura analógica da el valor mas alto (1024) y para que nos de un valor de verdad necesitamos una resistencia de 100 KΩ que se conectará desde la patilla de Out a la patilla de Gnd y las conexiones son iguales que las del sensor anterior, conectamos a tierra GND, conectamos Out al zócalo de entradas analógica número 3 del arduino (ya que el programa del proyecto lee de ahí la temperatura, estaros atentos ya que si falláis en la conexión os saldrá unos datos sin sentido, ejemplo 2000 grados de temperatura) y ya después la ultima patilla la conectamos a 5V para alimentar el sensor.

Ya con esto tenemos el montaje físico de nuestras Motas (Mota = Arduino + Sensores).
Si tenéis alguna duda en la web del proyecto, podéis visitar el tutorial que está colgado en la web, http://proyecto-canyafote.homelinux.org/tutorialMontaje.html y véis el montaje desde un método más visual.

Paso 2: Comencemos a subir los programas a los arduinos

Antes de definir que rumbo llevar debemos descargarnos el IDE de arduino para modificar, subir, etc... los sketch o programas de los arduinos, lo podéis descargar desde está web http://arduino.cc/en/Main/Software , esta página web no sirve solo para descargaros el IDE, si no que si por ejemplo queréis modificar el código del proyecto podéis encontrar tutoriales para

aprender a programar para arduino o podéis visitar otros proyectos que estén desarrollando con arduino.
Bueno una vez descargado el IDE (es muy recomendable que sea la última ya que las librerías cambian y es bueno estar al día de cada una de ellas), lo descomprimimos y ya tenemos nuestro IDE de arduino. Ahora antes de ver como subir los sketch o programas de los arduinos, os voy a dar un pequeña explicación para que aprendáis a usar este programa.
Lo primero antes de todo es ejecutar IDE del arduino, para ello buscamos la carpeta que nos ha aparecido al descomprimir el archivo que nos hemos bajado antes, el nombre será algo como "**arduino-XXXX**" (el XXXX es el número de la versión que os habéis descargado), y buscamos el ejecutable que se llama "**arduino**".
Al ejecutarlo nos saldrá una pantalla tal que así

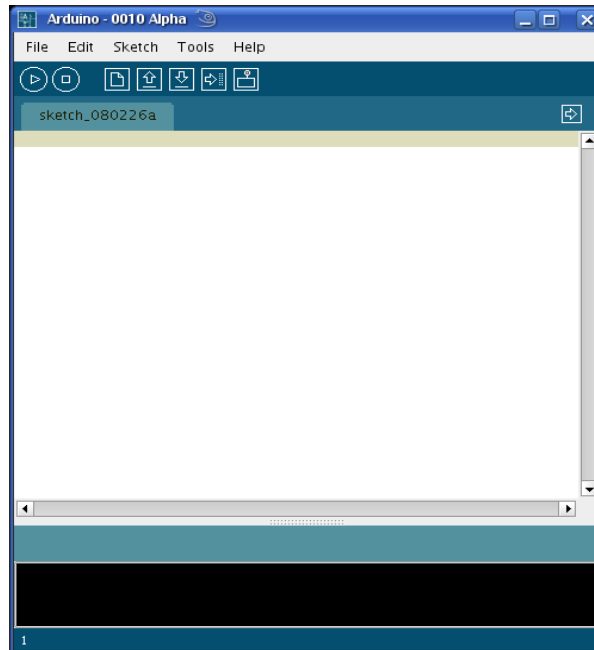


Foto del IDE de arduino

Está va a ser nuestra herramienta de trabajo cuando utilicemos los arduinos, voy a explicaros un poco como dividirla, se podría dividir en tres partes, la primera parte es la parte de los botones que más adelante os explicaré el funcionamiento de cada uno, la segunda parte, es la parte blanca de la foto anterior que es la zona de código, y la tercera es la parte de abajo (cuadro negro) que es una pequeña consola, que te enseña los Log, ejemplo: si el programa ha sido bien compilado o si no lo ha sido porque ha fallado, etc...
Paso a explicaros los botones principales o los que vamos ha utilizar mucho:

Botón de Play: Sirve para comprobar si nuestro Sketch tiene algún fallo de sintaxis, ejemplo: que nos falte un ; , o que no declaremos un atributo, etc...

 Botón de Stop: Sirve para parar la comprobación, por si has recordado que te faltaba algo.

 Botón de New: Al pulsarlo te habré una nueva ventana vacía para empezar a crear otro sketch.

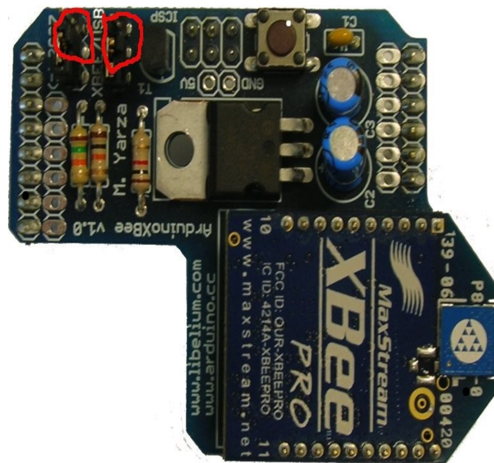
 Botón de Open: Sirve para abrir algún otro sketch que tenemos guardado o descargado.

 Botón de Save : Este botón va grabando lo que llevemos hecho en el IDE como un archivo .pde que es el formato que usa este IDE.

Botón de Upload: Sirve para subir el Sketch que tenemos en la pantalla del IDE al arduino, además este botón hace una comprobación con el botón Play antes de subirlo.


Botón Terminal: Aparece un pequeño terminal, donde si en nuestro programa hemos usado la función Serial, nos aparecerá escrito la frase que hemos dicho que nos imprima.

Ahora hay que diferenciar entre el montaje del módulo Ethernet o el módulo Xbee, voy a empezar con el módulo Xbee, ya que hay que hacer un par de pasos diferentes al módulo Ethernet.
Antes de comenzar debemos configurar nuestros módulos de Xbee, para ello debéis visitar la web, ya que tengo la configuración de estos módulos paso a paso, http://proyecto-canyafote.homelinux.org/configxbeemodulo.html .
Después de configurar este módulo, ya tenemos configurados nuestro módulo receptor y nuestro módulo escritor, debemos subir los programas y para subir los programas debemos colocar los pines en modo USB, lo debemos desplazar a la zona donde está dibujada en rojo (un poco mal por cierto)



Estaros atentos a este paso anterior, ya que si no lo hacéis no podréis subir los programas, ya que os dará errores en la consola de errores del IDE. Vamos a pasar a subir los programas de los arduinos que tienen los módulos Ethernet, ya que tienen una pequeña configuración.

Lo primero que tenemos que hacer es ejecutar el programa IDE de Arduino, cuando ya este ejecutado, debemos conectar el arduino al ordenador con el cable Usb, una vez conectado vamos a la pestaña *Tools → Serial Board → Elegimos el número de nuestro Usb.* Y después debemos decirle al IDE el tipo de placa que tenemos, por ejemplo yo utilizo Arduinos Duemilanove, por tanto debo ir a *Tools → Boards → Elegiré el tipo de placa (ejemplo Duemilanove).*
Ya con esto debemos abrir el programa que nos hemos descargado (Sketch programa escucha, Sketch programas Xbee lectura o Sketch Reaccion), para abrirlo debemos darle al botón y navegar en la ventana que nos aparece, para encontrar alguno de los programas que he nombrado antes y haced doble click sobre él. Una vez abierto solo debemos modificar un par de cosas, os muestro las líneas que se tiene que modificar.

```
//Mascara del arduino, cambiar si se desea, o cambiarla si se tienen varios arduinos
byte mac[] = { 0xDE, 0xAD, 0xB1, 0xE2, 0x1E, 0xED };
// Ip de los arduino, modificarlo entre ellos
byte arduinoip[] = { 192, 168, 0, 115 };
// Ip del servidor, colocarle la ip del ordenador al que quieren que le envie la
informacion
byte ip[] = { 192, 168, 0, 100 };
Client client(ip,9999); //Decimos que trabaje como client
```

Estas son las líneas que debéis modificar ya que si no son arregladas no os va a funcionar el envío de datos desde el arduino al servidor. La primera línea azul es la mascara que le vamos a dar, podéis dadle los valores que queráis mientras que sean en Hexadecimal, y con la estructura que veis. La segunda línea, da la ip al arduino, podéis dadle la ip que queráis pero mirar que no se os solape con ningún arduino ni con el servidor, ya que si no crearía conflicto y no se conectaría (si queréis comprobar si estos dos pasos han ido bien, cuando terminéis de configurar entero el arduino, conectarlo a la red y si se iluminan 3 luces amarillas está bien, si no se iluminará una que pone HREF, que significa que no tiene ip para direccionar), una vez dado ip al arduino, debemos direccionar los arduinos a nuestro servidor, para ello debemos obtener la ip del servidor, en mi caso es 192.168.0.100 (ya que es una ip local de mi router) ustedes debéis dadle el valor de vuestro servidor o del ordenador donde queréis que se envíe la información. Y ya por ultimo, debéis colocar el puerto por donde queréis que lleguen los datos, en mi caso es el 9999, podéis colocar el que queráis pero debéis recordadlo ya que para pasos siguientes vamos a necesitadlo (deciros que tengáis cuidado con una serie de puertos ya que hay algunos que están configurados para montar paginas web, servidores de correo, y si queréis en un futuro haced una página web para los datos que obtenéis os dara problemas si los arduinos tienen el puerto 80 por ejemplo). Otra modificación que debéis hacer es que en el programa del arduino busqueis estas líneas:

```
    // Efectuamos el envio al servidor con toda la informacion, donde se envia la informacion para construir el xml en el servidor
        sprintf(buf, "HTTP/1.1 200 OK\n"
                "Content-Type: text/plain;charset=UTF-8\n"
                "Content-Length: 49\n\n"
                "<xml><entry><sensor>1</sensor><humedity>%s</humedity><temperature>%s</temperature></entry></xml>", humedad_str, temperatura_str);
```

Si tenéis varios arduinos captando, sería bueno que cambies el número de identificación porque si no tomará todos los datos como si fueran del mismo arduino, por ejemplo yo tengo dos, uno es el 1 y otro el 2.

Ahora solo nos falta subir el programa al arduino, para subirlo, es conveniente que antes se compruebe a ver si se os ha olvidado algo al modificar lo que yo os he dicho. Para ello dadle al botón de play,

y os saldrá una frase "*Binary sketch size: * bytes (of a * byte maximum)*" y arriba "*Done compiling"* Si os aparece esta dos frases es que el programa esta perfectamente, si os aparece otra cosa, leed bien que os dice y comprobar que lo que habéis modificado está con la estructura del ejemplo que os he puesto. Y ya solo nos falta subidlo a la placa, para subid el programa debéis dadle al botón Upload , y os aparecerá los mensajes que os he dicho antes otra vez.

Ya con esto tenemos configurado los módulos Ethernet, el único programa que queda por explicar es el "Sketch programa de Escritura Xbee" que debéis conectar el arduino al ordenador, elegid el tipo de placa y el usb (*Tools → Serial Boards* .....) como en los pasos anteriores, y solo necesitáis buscar la linea está:

```
        // construimos el paquete que vamos a enviar
    uint8_t payload[] = { 2, temperature, humedity };
```

Igual que en el Ethernet, si tenéis varios arduinos Xbee sería recomendable que le cambiéis el número de identificación, por ejemplo donde pone el 2 en la caja, id cambiándolo, por ejemplo si tenéis tres, modificad en cada uno el 2 de la línea anterior por un 1, en otro lo dejas el 2 y el último

colocad un 3 por el 2, y al servidor se guardará información separada de los 3 arduinos que serviran para generar gráficas independientes en el siguiente paso. Solo falta que le deis al botón Upload. Después de estos pasos de configuración y de subida de programas, ya solo falta configurar los programas de escucha del servidor, si tenéis duda en la parte de tutoriales de la página web, podéis ver una par de videos, os los adjunto:

http://proyecto-canyafote.homelinux.org/tutorialSubir.html : Video para subir los programas
http://proyecto-canyafote.homelinux.org/tutorialconfigxbee.html : Video para configurar Xbee

Paso 3: Configurar Programa Escucha del Servidor

Lo primero que tenemos que hacer es encontrar la carpeta de "Programa de Escucha" y abris con el programa "SocketServidor.py" con cualquier editor de texto, (si estáis en windows os recomiendo Notepad ++) una vez abierto vamos a buscar estas líneas:

```
def RecogiendoInfo():
        # esta funcion sirve para que cuando detecta el cierre del programa forzado
efectue la funcion handler, que sera nuestro tratamiento de error
        signal.signal(signal.SIGINT, handler)
        print 'Iniciando el programa'
        server = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        # creamos el socket servidor
        server.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
        # cambiamos las opciones para que el socket sea liberado cada vez k se usa
        server.bind(('0.0.0.0',9999))
```

La única modificación que debéis hacer es abajo en la línea "server.bind(('0.0.0.0',9999)) el número 9999 debéis modificarlo por el número de puerto que le habéis dado a los arduinos, si por ejemplo le habéis dado el 6787, en vez del 9999 debéis escribir 6787. y ya tenemos configurado nuestro Programa de escucha, ahora debéis guardar y cerrar el editor de texto.
Ahora solo falta ejecutar el programa, que lo efectuaremos a partir de una terminal, usando el comando "python SocketServidor.py" al efectuar esto, el programa se ejecutará y esperará a recibir los datos y a guardarlos en los xml.

Paso 4: Como generar las gráficas

Este paso no necesita ni configuración ni nada, lo único que tenéis que hacer es coger el contenido de la carpeta del Parser que queráis usar, ejemplo : Queréis usar el Parser Interdia, pues copiáis el contenido de la carpeta de Parser Interdia, y esto es muy importante, todos lo que está dentro de la carpeta Parser Interdia y lo copiáis en la carpeta del Programa de Escucha. Ya lo único que queda es que escribáis "python ParserInterdia.py *.xml" **(*** para que utilice todos los archivos .xml si lo quieres usar de uno en especial le dices el nombre seguido de .xml y os aparecerá unas gráficas como estas.

Gráfica Media

Gráfica Interdia

Dudas o Cuestiones

Si tenéis dudas, cuestiones o sugerencias, podéis enviar un correo a proyecto.canyafote@gmail.com y os contestaré con mucho gusta vuestas dudas.